

PART 1:

1.1) NAME OF ORGANIZATION:

PRERNA SEWA SANSTHAN

ADDRESS & CONTENT INFORMATION OF PRERNA SEWA SANSTHAN:

CENTRAL OFFICE: D-152, SECTOR-23

SANJAY NAGAR GHAZIABAD-201002

UTTAR PRADESH, INDIA

MOB:- +919891069636

EMAIL:- PrernaSewa2007@gmail.com

MISSION OBJECTIVE OF THE ORGANISATION:

IT IS AN NGO WORKING TOWARDS THE UPLIFTMENT OF POOR & OLD HOMELESS PEOPLE

ENVIRONMENT IN WHICH THEY OPERATE IN:

THEY ARE BASED IN GHAZIABAD, INDIA BUT THEIR OPERATIONS EXTEND TO NEIGHBOURING CITIES AS WELL

STRUCTURE OF ITS LEGISLATION:

PRERNA SEWA SANSTHAN WAS ESTABLISHED IN 2011 AND IS REGISTERED UNDER INDIAN REGISTRATION TRUST ACT 1882. IT IS ALSO REGISTERED IN SECTION 80G & 12A OF INCOME TAX ACT 1961 WITH INCOME TAX DEPARTMENT.

1.2) DEFINING THE SOME OF THE PROBLEMS DISCOVERED FROM THE ORGANISATION:

The NGO is currently facing the issue of data management and these are poor system tracking in regards to:

- Tracking the donations which they receive
- Tracking the social workers working for them
- Tracking the homeless people taking shelter in their home
- Lastly would be tracking the food & medicine inventory they for helping the homeless

1.3) EXPECTED OUTCOMES OF THE PROJECT:

Keeping track of what goes in and out at Prerna Sewa Sansthan home for the homeless through manually using paper folders and file registers can be error prone and so to help the organisation work at the best of their ability and to help them also help the homeless people effectively they will need to manage their data properly. I will need to help them come up with a database that will help them achieve data management efficiently and one that will help them track well. Being an NGO they should opt for a free RDBMS or one with a less price which will help save money needed for even more important tasks.

1.4) PART OF THE PROJECT & BOUNDARIES PERMISSION:

Database will hold tables (relations) for each of the following entities: Donations, inventory, Employees (including social workers), shelters, and residents (include poor and old homeless people in shelters). They will also require a simple front-end (web-based or local) to enter and retrieve data. The insertion, updating, election and generation of reports can be easily done via SQL statements. Using a database can help reduce the need to handle multiple hard registers and file folders. The manual effort and time spent in going through the data, keeping track of it and the insertion/correction work will highly be reduced (as data manipulation and retrieval is done within seconds using database). Reports generated would be much more accurate than the ones generated manually. This all would reduce their cost and manual effort/time and will help them work and make decisions more efficiently. This will in turn, help them help others better.

PART 2:

2.1) SOURCE OF INFORMATION:

I did this research based on a case study of Prerna Sewa Sansthan by searching through their about page which talks about how they track about what goes in and out of the organization itself and some of their ratings from Facebook, Twitter and Instagram.

2.2) LINKS FROM WHERE I COLLECTED THE INFORMATION FROM:

Some of their information was taken from the Prerna Sewa Sansthan website <http://prernasewa.org/> and <http://www.doaram.com/organization/prerna-sewa-sansthan-trust> which is their trust website that gives more in details about and more about their donations and how one can donate to them on <https://urgesture.com/prernabalashram>

- Who they are in the community
- The general statistics about the residents whom are taken care of in the centre
- Statistics of some of the abandoned animals that they take care of
- Education that's offered to the homeless
- More about their rural development

From the website below is their Facebook reviews https://web.facebook.com/PRERNASEWASANSTHAN/reviews/?_rdc=1&_rdr which talks a lot more about some of the reviews from those that work there and some that get to review from the organization itself which had then drove me to my conclusion of their struggle of using paper-based folders and file registers that have become error prone to track it's donations, the residents staying at the home, their employees, some of their shelters in the community itself and their inventories

2.3) RULES OF THE ORGANIZATION:

- Prerna Sewa Sansthan, the parent NGO under which this Sewa Initiative runs. Is a 80G registered NGO. They help the kids who have either been abandoned or their parents are not in the right position of taking care of them.
- They started off with 30 kids and some of the rules laid there are that once the parents are in good condition to take care of the kids, they hand them over or if they reach a working age, they let them go.

-There are different ways in which people can help them achieve their vision. They can donate to them in cash through following some of their links and another one on how you can go about it if wanting to volunteer

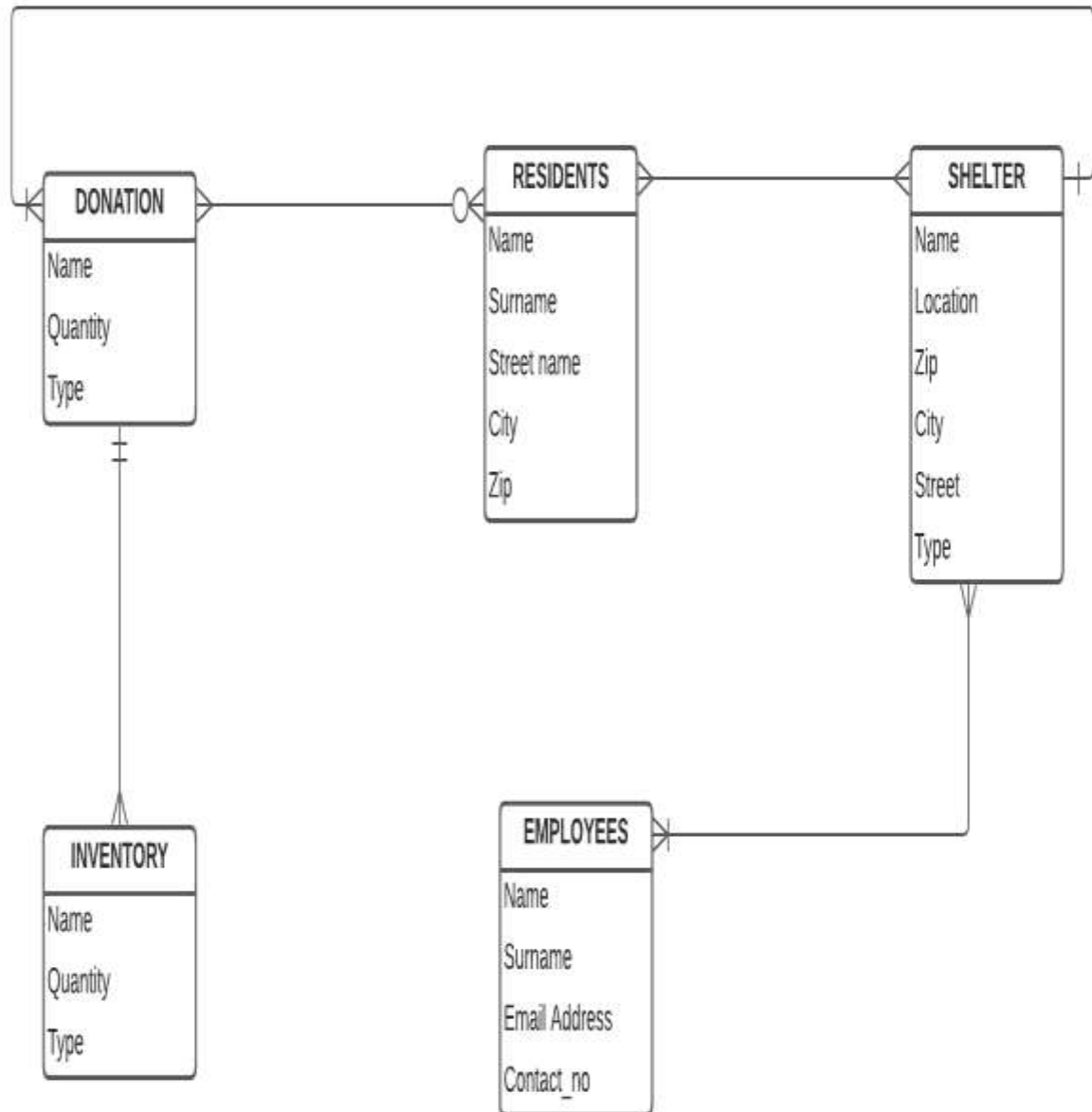
STEP 3:

3.1) the database design approach I will be using is the RDBMS (Relational Databases Management System). This is a database management language known to offer a highly organized and structured approach to information management.

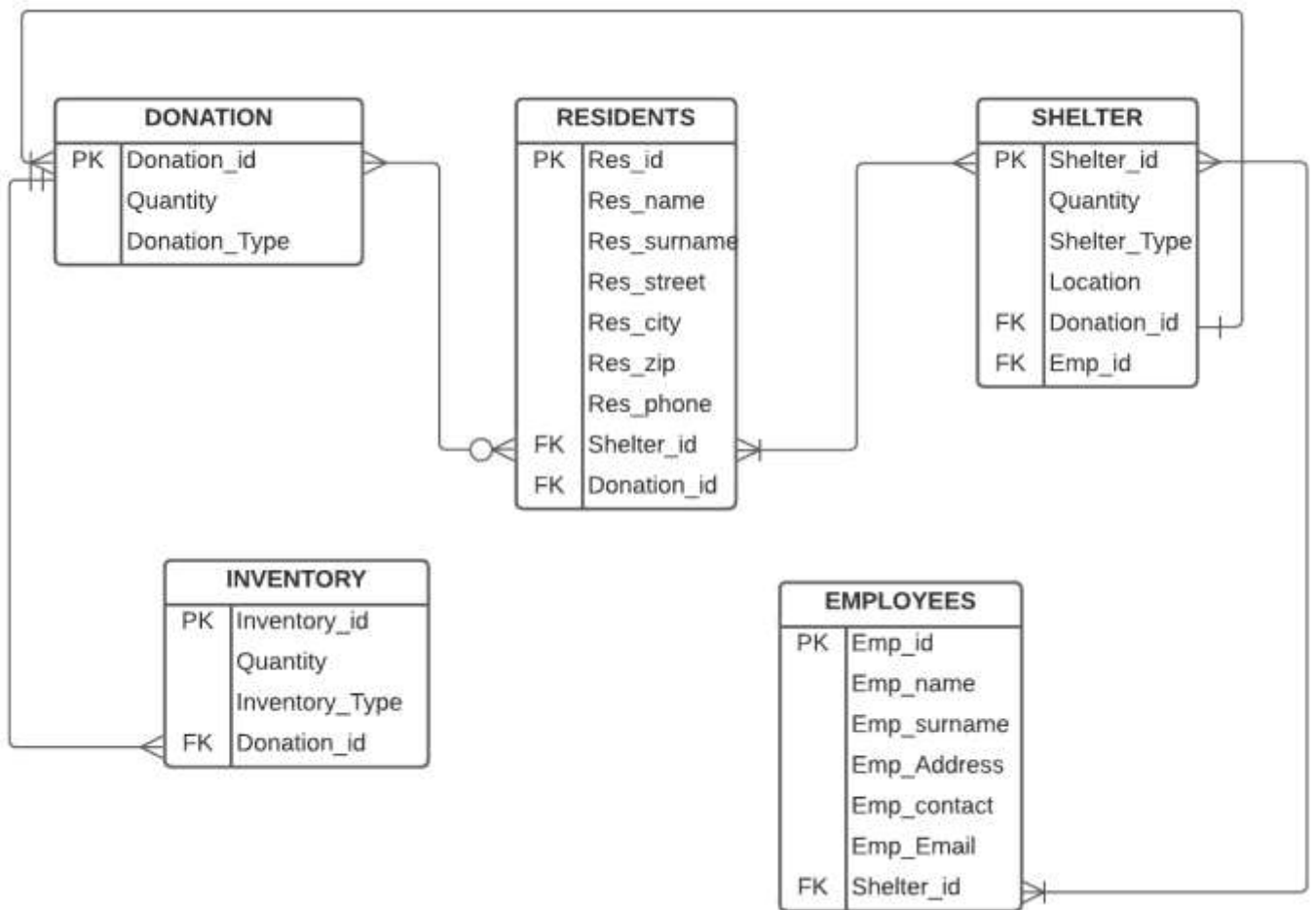
REASON:

- Satisfies set of priorities that measure the atomicity, consistency, isolation and durability of database systems.
- Ideal for consistent data systems. It is very good if you need a dynamic information system to measure amounts of data and not dealing with numerous data types
- Offers great speed and stability
- And since it has been up for 40 years now, it is also easier to get support, add-on products and lastly integrates data

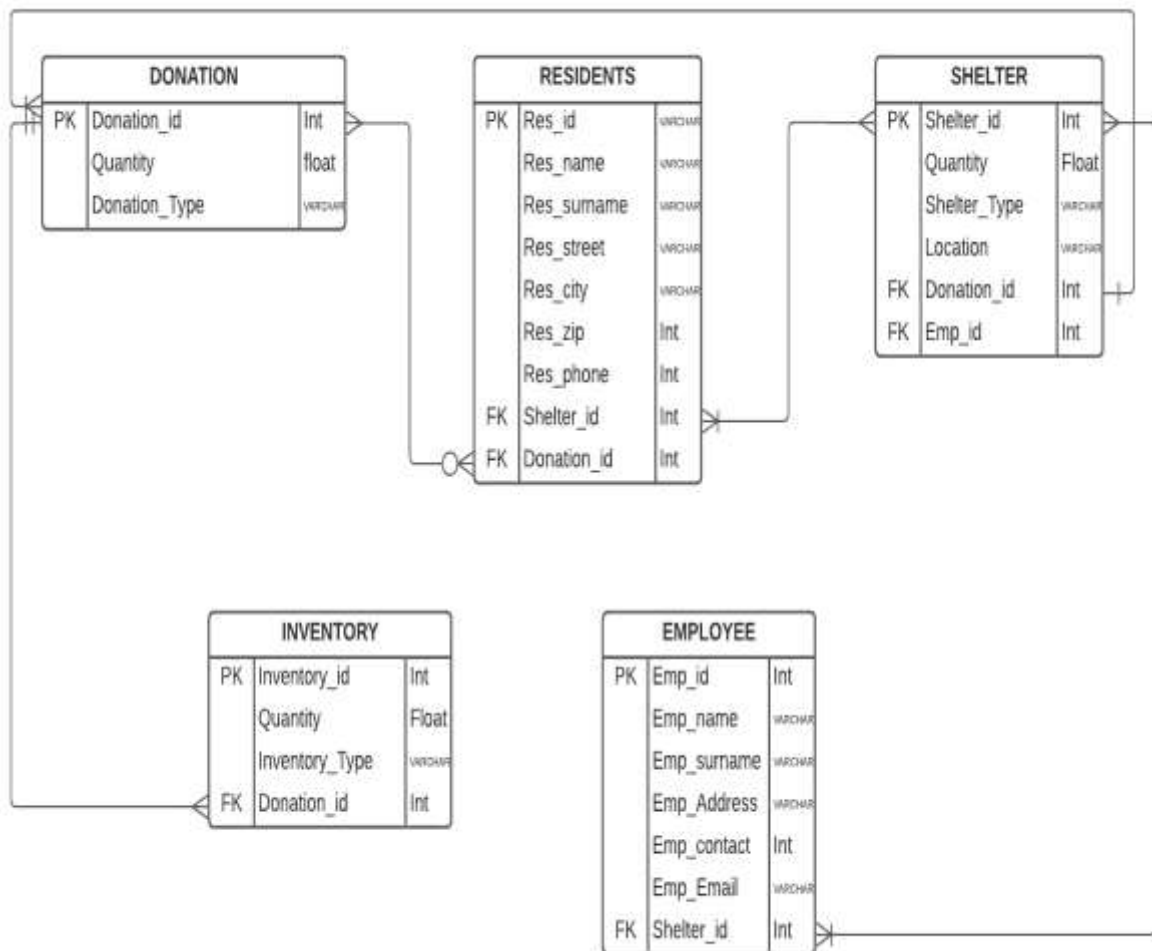
3.2) CONCEPTUAL DESIGN:



3.3) LOGICAL MODEL:



PHYSICAL DESIGN:



PART 4:

4.1) REQUIREMENTS OF SIMPLE FRONT-END(WEB-BASED OR LOCAL) AND HOW TO ENTER & RETRIEVE DATA:

- Once we have created the database and our initial tables, we can insert data into them. We then can insert data into them, then interact with the data management module which manages the physical storage of data and works closely with the data storage module
- The module allows the developer to do a data lookup operation through SELECT statements, data modifications through UPDATE statements and data removal through DELETE statements, among other things.
- Like DDL operations for instance, these are the first type of operations that one will execute on their database through the RDBMS once it is installed. DDL stands for “data definition language”. It is a list of SQL commands that one can use not only to create the first objects in a database but also make changes. Also is not limited to creating DDL statements begin with some common keywords:
 - Create : create any type of database eg. Tables.
 - Drop : Used to delete any database including tables, users, functions etc.
 - Filter : Used to modify any object in database
 - Truncate : Is used to remove all data from the table.
- Each transaction must display the ACID test concept which is:
 - “A” – stands for atomicity. It guarantees that either all the changes made by a user are applied to the table or no changes are applied at
 - “C” - stands for consistency. It guarantees that the execution of a transaction keeps the database and its data in a consistent state.
 - “I”- stands for isolation. It allows for multiple transactions to execute at the same time with transparency to other users, provided they do not change the same data
 - “D” – stands for durability. It ensures that once a transaction is committed and applied, it will remain committed even if the database shuts down

4.2) Formulate possible SQL statements for the Database:

This is where insertion, updating, election and generation of can be easily done:

Conceptual design:

```
CREATE TABLE `DONATION` (
  `Name` <type>,
  `Quantity` <type>,
  `Type` <type>
);
```



```
CREATE TABLE `RESIDENTS` (  
  `Name` <type>,  
  `Surname` <type>,  
  `Street name` <type>,  
  `City` <type>,  
  `Zip` <type>  
);
```

```
CREATE TABLE `SHELTER` (  
  `Name` <type>,  
  `Location` <type>,  
  `Zip` <type>,  
  `City` <type>,  
  `Street` <type>,  
  `Type` <type>  
);
```

```
CREATE TABLE `INVENTORY` (  
  `Name` <type>,  
  `Quantity` <type>,  
  `Type` <type>  
);
```

```
CREATE TABLE `EMPLOYEES` (  
  `Name` <type>,  
  `Surname` <type>,
```

```
`Email Address` <type>,  
`Contact_no` <type>  
);
```

Logical Design:

```
CREATE TABLE `DONATION` (  
  `Donation_id` <type>,  
  `Quantity` <type>,  
  `Donation_Type` <type>,  
  PRIMARY KEY (`Donation_id`),  
  FOREIGN KEY (`Donation_id`) REFERENCES `SHELTER`(`Donation_id`)  
);
```

```
CREATE TABLE `RESIDENTS` (  
  `Res_id` <type>,  
  `Res_name` <type>,  
  `Res_surname` <type>,  
  `Res_street` <type>,  
  `Res_city` <type>,  
  `Res_zip` <type>,  
  `Res_phone` <type>,  
  `Shelter_id` <type>,  
  `Donation_id` <type>,  
  PRIMARY KEY (`Res_id`)  
);
```

```
CREATE TABLE `INVENTORY` (  
    `Inventory_id` <type>,  
    `Quantity` <type>,  
    `Inventory_Type` <type>,  
    `Donation_id` <type>,  
    PRIMARY KEY (`Inventory_id`),  
    FOREIGN KEY (`Donation_id`) REFERENCES `DONATION`(`Donation_id`)  
);
```

```
CREATE TABLE `EMPLOYEES` (  
    `Emp_id` <type>,  
    `Emp_name` <type>,  
    `Emp_surname` <type>,  
    `Emp_Address` <type>,  
    `Emp_contact` <type>,  
    `Emp_Email` <type>,  
    `Shelter_id` <type>,  
    PRIMARY KEY (`Emp_id`)  
);
```

Physical Design:

```
CREATE TABLE `SHELTER` (  
  `Shelter_id` Int NOT NULL AUTO_INCREMENT,  
  `Quantity` Float,  
  `Shelter_Type` VARCHAR(40) NOT NULL,  
  `Location` VARCHAR(100) NOT NULL,  
  `Donation_id` Int,  
  `Emp_id` Int,  
  PRIMARY KEY (`Shelter_id`)  
);  
  
CREATE TABLE `DONATION` (  
  `Donation_id` Int NOT NULL AUTO_INCREMENT,  
  `Quantity` float,  
  `Donation_Type` VARCHAR(40) NOT NULL,  
  PRIMARY KEY (`Donation_id`),  
  FOREIGN KEY (`Donation_id`) REFERENCES `SHELTER`(`Donation_id`)  
);  
  
CREATE TABLE `RESIDENTS` (  
  `Res_id` Int NOT NULL AUTO_INCREMENT,  
  `Res_name` VARCHAR(40) NOT NULL,  
  `Res_surname` VARCHAR(40) NOT NULL,  
  `Res_street` VARCHAR(100) NOT NULL,  
  `Res_city` VARCHAR(100) NOT NULL,
```

```
`Res_zip` Int,  
`Res_phone` Int,  
`Shelter_id` Int,  
`Donation_id` Int,  
PRIMARY KEY (`Res_id`)  
);
```

```
CREATE TABLE `INVENTORY` (  
  `Inventory_id` Int NOT_NULL AUTO_INCREMENT,  
  `Quantity` Float,  
  `Inventory_Type` VARCHAR(100) NOT NULL,  
  `Donation_id` Int,  
  PRIMARY KEY (`Inventory_id`),  
  FOREIGN KEY (`Donation_id`) REFERENCES `DONATION`(`Donation_id`)  
);
```

```
CREATE TABLE `EMPLOYEE` (  
  `Emp_id` Int NOT_NULL AUTO_INCREMENT ,  
  `Emp_name` VARCHAR (40) NOT NULL,  
  `Emp_surname` VARCHAR (40) NOT NULL,  
  `Emp_Address` VARCHAR (100) NOT NULL,  
  `Emp_contact` Int,  
  `Emp_Email` VARCHAR (40) NOT NULL,  
  `Shelter_id` Int,  
  PRIMARY KEY (`Emp_id`)  
);
```

PART 5:

5.1) Different types of database performance turning methods & the one that I used:

MySQL performance turning is the process of maximising query speeds on a relational database and these include things like:

Database Query Optimization Modes:

It is based on internal storage & aggregation data that is collected by the OLAP process and what it also contains is the **Automatic query optimization, manual query optimization, static query optimization and dynamic query optimization.**

DBMS Performance Turning:

This describes a group of activities used to optimize and homogenize the performance of a database. It usually overlaps the query tuning, but refers to design of the database files, selection of the database management system application and configuration of the database's environment and this includes things like the **SQL performance tuning and DBMS performance tuning**

DBMS Architecture:

This is a representation of DBMS design. It helps to design, develop, implement, and maintain the database management system. It allows dividing the system into individual components that can be independently modified, changed, replaced and altered. These include such things like the **SQL cache and Input/output request.**

5.2) The type I used is the SQL performance turning:

The reason being is the its optimization is usually an interactive approach to making and then monitoring modifications to an application and it's database.

How does it work?

- Avoid SELECT*. This tip is particularly important if you have a large table (think hundreds of columns and millions of rows). If an application only needs a few columns, include them individually instead of wasting time querying for all the data. Again, reading extra data will cause CPU utilization to spike and memory to be trashed. You should check the page of life expectancy to make sure that you are not having such an issue.
- Check indexes
- Work with the smallest data set required

- Remove calculations in your JOIN & WHERE clauses.
- Use wildcards only at the end of the phrase
- When searching for plaintext data (i.e. Cities or names) try to use wildcards sparingly to search large amounts of data.
- Run query day off-peak hours
- Use WHERE instead of defining filters

CHALLENGES ENCOUNTERED AND HOW TO GO ABOUT FIXING THOSE THE NEXT TIME:

Some of the challenges encountered were finding an institution that allowed me to do thorough investigations on their premises. I sent numerous of emails to schools around me; even tried calling to set an appointment but none of them had gotten back to me. I even got kicked out of one school premises, telling me to set an appointment online, the very same school that never picked up any of my calls or returned to any my emails. As for a retirement home near me, they also had not gotten back to me

Another challenge I had encountered in the case study was finding the relevant reviews from people online to help me reach a conclusion on sectors that need to be improved. As from far, they seemed to be perfect till I conducted a deep search of this particular home for the homeless in Ghaziabad.

How can the assignment be improve/ how UNISA can help intervene in this matter:

I think the best way is for UNISA to provide us with links that will help direct us to these real life case study or even help provide us with some of the case studies to help us choose from to help us conduct the research if possible.